

REMARKS/ARGUMENTS

Prior to entry of this amendment, the application included claims 1-6 and 8-25. No claims have been amended, cancelled or added. Hence, after entry of this Amendment, claims 1-6 and 8-25 stand pending for examination.

Rejections Under 35 U.S.C. § 103(a)

Claims 1-6, 8-12 and 25 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent Publication No. 2002/0129129 to Bloch et al. (“Bloch”). Claims 13-24 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Bloch in view of U.S. Patent No. 6,314,451 to Landsman et al. (“Landsman”). Applicant traverses.

Generally speaking, the claims recite at least a system or method that receives a plurality of text files at a client computer system. A program resident on the client computer system executes and creates an application from the downloaded text files that is executable independent from the program that created it. The claims may include various other elements and/or limitations beyond these.

In order to establish a *prima facie* case of obviousness, the Office Action must establish: 1) some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the references or combine their teachings; 2) a reasonable expectation of success of such a modification or combination; and 3) a teaching or suggestion in the cited prior art of each claimed limitation. See MPEP §706.02(j). As will be discussed below, the reference cited by the Office Action does not teach or suggest each claimed limitation. For example, Bloch does not teach or suggest an application that is created from downloaded text files and that is executable independent from the program that created it. Instead, Bloch teaches an application that is executable only within the program (the AVM) that created it.

For instance, paragraph [0028] of Bloch describes the AVM as providing “a method for deploying and executing Extensible Markup Language applications . . .” Paragraph

[0030] states that “[t]he AVM interprets XML and other code downloaded from a web server on the network computer to render a full-featured application on the client computer with a native look and feel for the client operating system platform on which it is installed.” Paragraphs [0032] and [0033] describe the AVM as providing a GUI interface for XML applications. Paragraph [0034] notes that “[t]he AVM is capable of assembling and executing a robust application . . .” Paragraph [0044] states that the AVM may be used “for deploying and implementing a software application . . .”

Most striking, paragraph [0047] states that “the AVM is used to execute ‘Virtual Applications’ that are assembled from files and programs residing on one or more computers on a network. Only AVM 221 is installed on a Client Device 10. Other files residing on Network Server(s) 131 are downloaded over a Network 121 to assemble the complete Virtual Application 20 that is then executed by the AVM 221 on the Client Device 10.” Paragraph [0047] goes on to note that “[t]he XML Files 144 and Image Files 145 specify the appearance and behavior of the assembled Virtual Application 20 during execution by the AVM 221 and in response to user interactions during such execution.”

It is clear from the language in Bloch that the downloaded application is not executable independent from the AVM. Yet, despite what is described in Bloch, the Office Action erroneously points to various portions of Bloch that allegedly teach this element of the claims.

The Office Action points to paragraphs [0086] and [0087] as teaching an application that is executable independent from the program that created the application. Paragraph [0086] describes tasklists that include client tasks, conditional tasks and host tasks. Paragraph [0087] states that these “[t]asklists are executed by the AVM when an event occurs to which they are linked.” A tasklist cannot be both executed by an AVM and executable independent from the AVM.

The Office Action also points to paragraph [0100]. This paragraph describes the interaction of the application and the AVM, for example, paragraph [0100] describes “[a]s the user interacts with components, the AVM 221 detects, in a platform dependent way, changes to the visual appearance of the component that reflect the entry of data, the selection of a choice, putting focus on a component, and the like (step 92).” The paragraph also describes how an AVM session is ended after function is executed by the system handler. If the function (or application) is truly independent of the AVM, why then is the AVM terminated at the end of the function? Thus, the application and AVM are dependent applications rather than independent applications.

The Office Action also points to paragraph [0102] as describing message boxes that are unique to the operating system. These message boxes occur in response to some action by the AVM. Thus, these message box tasks are dependent on the AVM and not an application created by the AVM. Moreover, these messages boxes are, as noted by the Office Action, operating system procedures that are not an application created by the AVM from downloaded text files. The claims require that the same application that is created by the AVM execute independent from the AVM. Operating system procedures or calls have not been downloaded by the AVM.

The Office Action also points to paragraph [0103] as describing remote procedures. Remote procedures are procedures that occur remote from the client computer system and not at the client computer system. The claims require, in part, that a program (i.e., the AVM) create an application from text files on the client computer system and that the application is executable independent from the program. Simply put, remote calls are not executed on the client computer system. Moreover, these remote calls are not part of the application downloaded and created by the AVM.

The Office Action also points to paragraph [0105] of Bloch as referring to a database handler. This paragraph does not teach or suggest that the database handle is executable independent of the AVM. There is also nothing in Bloch to suggest that the database handler is created by the AVM from text files.

The Office Action also points to paragraph [0109] of Bloch as referring to “user interface . . . accessing any database . . . to test new software.” Paragraph [0109] describes how the AVM is used to test software, assemble user interfaces, access any database, etc. Therefore, the AVM as described is not independent from anything. It is completely dependent with software testing, user interface assembling and database accessing.

The Office Action fails to provide a *prima facie* case of obviousness. Simply put, there is nothing in Bloch that teaches or suggests an application that is executable independent from the program that created it from downloaded text files.

The Office Action responds to Applicant’s previous arguments by claiming that the claims and the specification do not provide implementation details showing execution independency. Office Action, pages 8-9. The Office Action then uses the specification to construe the meaning of “executable independent from said program.” However, construing claim language in light of the specification is improper during prosecution. See MPEP § 2111.01. The words of the claims must be given the ordinary and customary meaning according to a person of ordinary skill in the art. It is clear from the claims that the ordinary meaning of an application that is executable independent from the program that created it, at the very least, does not include an application whose execution depends on the program that created it as described by Bloch.

The Office Action also seems to suggest that there is an enablement issue with the claims. Office Action, page 8. The claims contain subject matter that is fully described in the specification in such a way as to enable a person skilled in the art to make or use the invention.

Conclusion

In view of the foregoing, Applicant believes all claims now pending in this application are in condition for allowance. The issuance of a formal Notice of Allowance at an early date is respectfully requested.

If the Examiner believes a telephone conference would expedite prosecution of this application, please telephone the undersigned at 303-571-4000.

Respectfully submitted,

Date: March 13, 2008

/Jason A. Sanders/

Jason A. Sanders

Reg. No. 59,984

TOWNSEND and TOWNSEND and CREW LLP
Two Embarcadero Center, Eighth Floor
San Francisco, CA 94111-3834
Tel: 303-571-4000
Fax: 415-576-0300
JAS/jln
61284548 v1